A Semi-Coarsening Strategy for Unstructured MG with Agglomeration

Jérôme Francescatto, Alain Dervieux

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

A Semi-Coarsening Strategy for Unstructured MG with Agglomeration

Jérôme Francescatto, Alain Dervieux

N° 2950

Juillet 1996

THÈME 4

ISSN 0249-6399

INRIA
SOPHIA ANTIPOLIS

A Semi-Coarsening Strategy for Unstructured MG with Agglomeration

Jérôme Francescatto[*] , Alain Dervieux[**]

Thème 4 — Simulation et optimisation
de systèmes complexes
Projet Sinus[***]

Rapport de recherche n° 2950 — Juillet 1996 — 27 pages

Abstract: The extension of the ideas of multigrid to complex compressible flow calculation is generally not straightforward, especially when non-embedded grids or volume agglomerations are used in order to work with unstructured grids. In this report, we study a multigrid method for solving second order PDE's on stretched unstructured triangulations. We use the finite volume agglomeration multigrid approach developed for solving the Euler equations. First, we present a method to generate coarse grids by volume agglomeration, allowing a directional semi-coarsening based on the Poisson's equation coefficients. The second order derivatives are approximated at each level by introducing a correction factor adapted to semi-coarsening. Then, we apply this method to solve the Poisson equation and extend it to the 2D turbulent Navier-Stokes equations with a low-Reynolds boundary heatment.

Key-words: Turbulent Flow - Compressible Flow - Multigrid Methods - Finite Elements - Finite Volumes - Semi Coarsening - Numerical Analysis

(Résumé : tsvp)

[*] E-mail: Jerome.Francescatto@sophia.inria.fr
[**] E-mail: Alain.Dervieux@sophia.inria.fr
[***] http://www.inria.fr/Equipes/SINUS-eng.html

Unité de recherche INRIA Sophia-Antipolis
2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex (France)
Téléphone : (33) 93 65 77 77 – Télécopie : (33) 93 65 77 65

Une stratégie d'agglomération anisotrope en multigrille et en maillage non structuré

Résumé : L'extension de l'idée multigrille pour le calcul d'écoulements complexes compressibles est généralement non triviale, spécialement quand les grilles sont non emboitées ou quand l'agglomération des cellules est utilisée sur des maillages non structurés. Dans ce rapport, nous étudions une méthode multigrille adaptée pour la résolution des EDP du second ordre sur des maillages étirés. Nous utilisons l'approche multigrille de type volumes finis agglomérés développée pour la résolution des équations d'Euler. Tout d'abord, on présente une méthode de génération automatique des grilles grossières par agglomération de volume, autorisant un semi-déraffinement directionnel, basé sur les coefficients de l'équation de Poisson. Les dérivées secondes sont approchées sur chaque niveau en introduisant un facteur de correction adapté au semi-déraffinement. On applique cette méthode pour résoudre l'équation de Poisson et ensuite étendue aux équations de Navier-Stokes couplées avec un modèle de turbulence à deux èquations $k - \varepsilon$ à bas Reynolds.

Mots-clé : Ecoulement Turbulent - Ecoulement Compressible - Méthode Multigrille - Eléments Finis - Volumes Finis - Semi Déraffinement - Analyse Numérique

Contents

Nomenclature

Symbol	*Description*	*Symbol*	*Description*
ρ	Fluid density	T	Temperature
u	Horizontal velocity	v	Vertical velocity
E	Total energy per unit volume	P	Pressure
k	Kinetic energy of turbulence	ε	Turbulent dissipation rate
x	X-coordinate	y	Y-coordinate
μ	Laminar viscosity	μ_t	Eddy viscosity
$\nu = \frac{\mu}{\rho}$	Kinetic laminar viscosity	$\mathcal{P}$	Eddy production term
τ	Laminar stress tensor	τ_t	Turbulent stres tensor
τ_w	Shear stress at the wall	$u_f = \sqrt{\frac{\tau_w}{\rho}}$	Friction velocity
Pr	Prandtl number laminar	Pt	Prandlt number turbulent
C_p	Specific heat at constant pressure	C_v	Specific heat at constant volume
$\gamma = \frac{C_p}{C_v}$	Specific heat ration	$c_f = \frac{2\ \tau_w}{\rho_\infty\ u_\infty^2}$	Friction coefficient
$Ry = \sqrt{\frac{\rho}{\rho_w}}\ \frac{\sqrt{k}\ y}{\nu_w}$	Local Reynolds number	$y^+ = \frac{u_f\ y}{\nu_w}$	Non-dimensional y

1 Introduction

The prediction of complex compressible flows modelled by the Reynolds-averaged Navier-Sokes modelling and unstructured meshes is getting some maturity. This maturity can be considered as reached when low-Reynolds statistical modelization (LRSM) such as the so-called low-Reynolds $k - \epsilon$ model, is easily computed in 2D and 3D. Many methods are becoming available for generating meshes including boundary layer severe stretching [11]. Also many improved numerical approximations are arising([4]). Our concern is the improvement of solution algorithms. Indeed, with the improvement of models, model error should be smaller, which requires the approximation error to be smaller; even with accurate approximations, this leads to using rather heavy meshes. As a result, the asymptotic complexity of the solution algorithm is important and this motivates to investigate the use of a multi-grid (MG) scheme.

A central issue in application of MG to LRSM is the efficiency of MG for highly stretched meshes. Indeed, nodewise (explicit multistep, Jacobi and Gauss-Seidel) smoothers are generally adopted for MG methods in flow problems. MG methods relying on these smoothers do not apply efficiently to stretched meshes since high frequencies aligned with the mesh are neither smoothed by the fine grid nodewise smoothing, nor by the coarse grid correction if full coarsening (in both mesh directions) is applied. The cures for this problem are of two kinds: the smoother can be improved up to a directionally implicit one such as a line relaxation, or the coarse mesh can be the result of a directional semi-coarsening. In this paper, we concentrate on a semi-coarsening strategy.

Starting with the option of treating stretched LRSM calculations by semi-coarsening, we have to decide how to semi-coarsen. In a lot of works dealing with structured meshes, several meshes are considered for each coarse level in order to account for all the possible semi-coarsenings at each level (see for example [22]). In the case of unstructured meshes, this is not very easy to adapt since a priori no special direction can be found in the mesh, except if we can identify the stretching direction. In this work, we propose to use a single coarsening, aligned with the stretching direction. This strategy has been studied for non-embedded meshes by [21]. For an approach in which volume-agglomeration is adopted for building coarser levels, a first study is presented in [18]. The present study aims at contributing to a deeper understanding of the mechanism of volume-agglomerated directional coarsening; in particular, we discuss the way the inconsistency introduced by the inaccurate transfers defined by agglomeration can be cured by an anisotropic correction factor.

For this purpose, we recall in a first section the motivation of directional or anisotropic coarsening, in order to predict how efficient it can be as compared with the application of the isotropic algorithm on isotropic meshes and on stretched meshes.

In a second section, we describe the anisotropic agglomeration process and the way the anisotropic correction factor can derived in a recursive way from one level to the coarser one.

Some first experiments are considered in section 3 and analysed for the scalar Poisson equation.

Then laminar Navier-Stokes flows are computed with an implicit formulation involving an agglomeration anisotropic MG algorithm.

At last, the main motivation of this study is condidered: a LRSM computation is presented and discussed.

2 Spatial approximation

The approximation developed in [5], is a combination of both finite-volume and finite-element methods. We assume that the computational domain Ω is bounded by a polygon. We introduce the following definitions : τ_h is a triangulation of Ω, n_h is the total number of vertices in τ_h and φ_i is the basis function associated with each node a_i.
We derive a dual finite volume partition of Ω, called the dual mesh of τ_h and made of the control volumes C_i built from triangle medians around each vertex a_i. (see Figure 1). The following definitions are introduced : $K(i)$ is the indices set of the neighboring nodes of a_i, $\partial C_{ij} = \partial C_i \cap \partial C_j = [G_{1,ij}, I_{ij}] \cup [I_{ij}, G_{2,ij}]$ is the interface between two cells and $\overrightarrow{n}_{ij} = \overrightarrow{n}_1 + \overrightarrow{n}_2$ is the approximation of the normal vector to interface δC_{ij}.

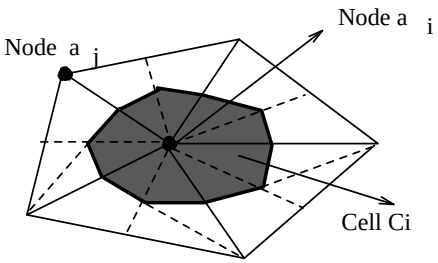

Figure 1: Median cell C_i

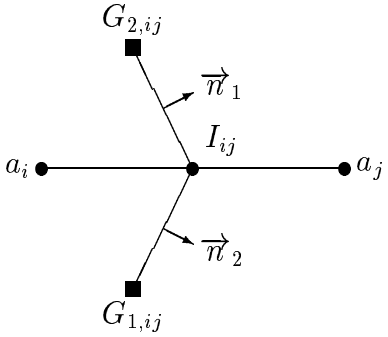

Figure 2: Interface ∂C_{ij} separating nodes a_i and a_j

We propose to discretize on $\Omega \subset I\!R^2$ the following advection-diffusion equation:

$$\begin{cases} -\triangle u + \overrightarrow{\nabla}.\left(\overrightarrow{V}u\right) = f, \quad \overrightarrow{V} \in I\!R^2, \quad u \in I\!R \\ u\,|_\Gamma = 0 \end{cases} \tag{1}$$

where Ω is an open bounded domain with a regular boundary Γ, $\overrightarrow{V}$ a propagation vector, f a continuous function and $u(x,y)$ a scalar function. The continuous problem on Ω, is approximated by the discrete problem defined on a triangulation Ω_h :

$$\underbrace{-\sum_{T, a_i \in T} \int\int_T \overrightarrow{\nabla} u.\overrightarrow{\nabla}\varphi_i \, dxdy}_{\text{diffusion}} + \underbrace{\sum_{j \in k(i) \,\cup\, i} \int_{\delta C_{ij}} u\, \overrightarrow{V}.\overrightarrow{n}\, d\delta}_{\text{advection}} = \int\int_{C_i} \chi(C_i)\ f_i \, dxdy \tag{2}$$

There are two different kinds of spatial discretization see [1] for details.
The *diffusive term* is calculated on each triangle and we use a standard P1-Galerkin formulation ("linear elements") :

$$\int\int_T \overrightarrow{\nabla} u.\overrightarrow{\nabla}\varphi_i \, dxdy = \sum_{j, a_j \in T} aire(T)\ u_j\ \overrightarrow{\nabla}\varphi_j.\overrightarrow{\nabla}\varphi_i$$

The *advective term* where the flux between the two cells C_i and C_j, is calculated through the interface ∂C_{ij}. The fluxes integration for i is obtained by summing the contribution of all

neighbors j. The integration on δC_{ij} of the advective term for $\overrightarrow{V}$ constant on all Ω_h is written by :

$$\int_{\delta C_{ij}} u\ \overrightarrow{V}.\overrightarrow{n}\ d\delta\ = \Phi\left(u_{ij}, u_{ji}, \overrightarrow{n}_{ij}\right) \tag{3}$$

where

$$\begin{cases} \Phi\left(u_{ij}, u_{ji}, \overrightarrow{n}_{ij}\right) = \alpha_{ij}\left(\ \theta_{ij}\ u_{ij}\ +\ (1-\theta_{ij})\, u_{ji}\ \right) \\ \alpha_{ij} = \int_{\delta C_{ij}} \overrightarrow{V}.\overrightarrow{n}_{ij}\ d\delta \\ \theta_{ij} = \dfrac{1}{2}\left(\ sign\left(\alpha_{ij}\right)\ +\ 1\right) \end{cases}$$

The values u_{ij} and u_{ji} are interpolates of u into interface ∂C_{ij}. The following choice results in a (at best) first-order scheme :

$$u_{ij} = u_i\ ;\ u_{ji} = u_j \tag{4}$$

A second order scheme is obtained by using a MUSCL interpolation :

$$\begin{cases} u_{ij} = u_i + \dfrac{1}{2}\left[(1-\beta)\left(\overrightarrow{\nabla} u\right)_{ij}^{cent} + \beta\left(\overrightarrow{\nabla} u\right)_{ij}^{upw}\right]\ .\ \overrightarrow{a_i a_j} \\ u_{ji} = u_j + \dfrac{1}{2}\left[(1-\beta)\left(\overrightarrow{\nabla} u\right)_{ji}^{cent} + \beta\left(\overrightarrow{\nabla} u\right)_{ji}^{upw}\right]\ .\ \overrightarrow{a_i a_j} \end{cases} \tag{5}$$

with

$$\left(\overrightarrow{\nabla} u\right)_{ij}^{cent}\ .\ \overrightarrow{a_i a_j} = u_j - u_i\ ;\ \left(\overrightarrow{\nabla} u\right)_{ij}^{upw} = \overrightarrow{\nabla} u\left(T_{ij}\right)\ ;\ \left(\overrightarrow{\nabla} u\right)_{ji}^{upw} = \overrightarrow{\nabla} u\left(T_{ji}\right) \tag{6}$$

where T_{ij} and T_{ji} are the triangles out of which and into which $\overrightarrow{a_j a_i}$ points as shown on Figure 3. The scheme is second-order accurate for $\beta = 1/2$ (scheme of Fromm) and is three-order accurate for $\beta = 1/3$.

Figure 3: Downstream and upstream triangles T_{ij} and T_{ji}

3 Linear multigrid method

The considered multigrid method is to be solved on the finest grid. It is an extension of the linear multigrid approach developed by M-H. Lallemand and A. Dervieux [15] to solve linearized Euler systems and extended to Poisson problems by B. Koobus and A. Dervieux [14], and to High Reynolds models by G. Carré [1].

3.1 Grid coarsening by isotropic agglomeration

The volume-agglomeration coarsening algorithm is based on a neighboring relation. From a fine unstructured triangulation, we first derive the dual finite-volume partition by building cells/volume around vortices that are made of part of median. we then generate automatically coarse finite-volume levels; to do this, we apply a technique of volume-agglomeration, described in [15] that assembles neighboring cells of the finest grid (e.g these having a common boundary) to build the cells of the coarser level, according to the following rule :

Consider successively every cell C_i of the domain.

[1] If C_i has already been included in a group (new coarse cell) then consider the next cell. Else create a new group containing C_i and put into this group neigbours of C_i which do not already belong to another existing group.

[2] Go to the next cell.

The main advantage of this method is to generate automatically one fastly the coarser grids without having to build a coarse genuine triangulation.

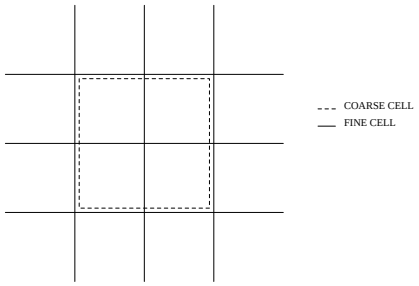

Figure 4: Isotropic agglomeration

3.2 Grid coarsening by anisotropic agglomeration

3.2.1 Why semi-coarsening ?

The essential principle of MG is to combine iterations that are good contractions for each subset of a partition of the whole set of frequencies of the considered system. In the case of an isotropic algebraic system, that is for example provided by a discretization on a uniform mesh of an isotropic system, it is well known that usual relaxation schemes are efficient contractions for all high frequencies. They can be successfully combined with a multigrid scheme relying on isotropic coarsening as sketched in Figure 4.

If some cause of anisotropy arises, such as either an anisotropic physical property, or a stretched mesh, then the above property is lost.
It is then possible to think of a new class of relaxation that would anyway smooth out all the high frequency modes; an example is line-relaxation; but such a choice is rather cumbersome to apply to unstructured meshes.
In this work we prefer to remark that usual relaxation is still a good smoother for a subclass of the high frequency modes, the *transverse* ones (to anisotropy direction), and to derive a coarser mesh that would still involve the non-smoothed *aligned* frequencies. This strategy is called semi-coarsening, and consists to increase the mesh size only in one direction. For the case of an isotropic problem discretized on a stretched mesh, this method is sketched in Figure 6. In the case of a periodic problem, with the Laplace operator, a local mode analysis shows that semi-coarsening in the adequate (transverse) direction does produce a fast converging multigrid algorithm when combined with a damped Gauss-Seidel relaxation whatever be the strength of anisotropy. It is in particular faster than the standard isotropic multigrid applied to an isotropic system. We illustrate this fact in Table 1 which presents reduction factors for both strategies; we observe that the best 2D case corresponds to the anisotropic algorithm applied to an infinitely anisotropic system, for which performances of the $1-D$ algorithm are obtained. However our strategy will involve an important difference as compared to structured-mesh strategies in which the semi-coarsening of one grid $(\Delta x, \Delta y)$ should produce *two* grids $((2\Delta x, \Delta y$ and $(\Delta x, 2\Delta y))$ in order to manage with any case of alignement. Indeed, in the proposed method, we assume that mesh stretching is the only source of anisotropy. Advective alignment is not addressed by semi-coarsening but by using enough transverse numerical viscosity. Thus, we may apply a *one-to-one* coarsening according to the detection and measure of the anisotropy of the fine mesh.

	Isotropic MG		*Anisotropic MG*	
ϵ	ω_{opt}	g_ϵ	ω_{opt}	g_ϵ
1.	0.80	0.60	0.80	0.60
0.5	0.91	0.82	0.71	0.43
10^{-1}	0.99	0.99	0.67	0.34
10^{-3}	1.00	1.00	0.66	0.33
10^{-5}	1.00	1.00	0.66	0.33

Table 1: Optimum relaxation parameter and reduction factor of the Jacobi method, according to $\epsilon = \triangle x/\triangle y$ and coarse

3.2.2 Anisotropic agglomeration technique

Our algorithm relies on two mechanisms. Firstly, we identify the stretching direction and strength, we call this, local metrics. Secondly, agglomeration is adapted to local metrics.

For building the local metrics, inspiring from an idea used in Algebraic Multigrid [15], we have chosen to measure the coefficients of the Finite-Element Laplace operator in order to evaluate the strong connections. We denote the matrix coefficients of A by a_{ij}, $(i,j) \in (1,\cdots,n_h)^2$.

Définition 3.1 *For a given cell C_i, we define the neighborhood of C_i by*

$$N_i = \{j \in I, \quad a_{ij} \neq 0\}$$

where we denote by $I = \{i, \quad i = 1, \cdots n_h\}$

Strong connections are defined by :

Définition 3.2 *We say that i is strongly connected to j if*

$$|a_{ij}| \geq \epsilon_1 \max_{p \in N_i} |a_{ip}|, \qquad \epsilon_1 = \frac{1}{4}.$$

We denote by S_i the set of indice j to which i is strongly connected.

Invariance by rotation of the Poisson equation allows to redefine for each node local rotated coordinates with basis $(\vec{e_\xi}, \vec{e_\eta})$. It enables to estimate a vector $\vec{V_i}$ indicating the stretching direction :

$$\vec{V_i} = \sum_{j \in S_i} |\overrightarrow{a_i a_j}.\vec{e_\xi}| \ \vec{e_\xi} + \sum_{j \in S_i} |\overrightarrow{a_i a_j}.\vec{e_\eta}| \ \vec{e_\eta} \tag{7}$$

in which a_i are cell barycenters, and where $(\vec{e_1}, \vec{e_2})$ is the Euclidean basis. The ratio L_i between components of $\vec{V_i}$:

$$L_i = \frac{\sum_{j \in S_i} |\overrightarrow{a_i a_j}.\vec{e_\xi}|}{\sum_{j \in S_i} |\overrightarrow{a_i a_j}.\vec{e_\eta}|} \tag{8}$$

determines the strength of stretching.
The *anisotropic agglomeration algorithm* is defined as follows :

Consider successively every cell C_i of the mesh.

[1] If C_i has already been included in a group (new coarse cell) then consider the next cell. Else create a new group containing C_i :
If $L_i \approx 1$, then put into this group neighbours of C_i which do not already belong to another existing group.
If $L_i \ll 1$ or $L_i \gg 1$, then put into together C_j such that $j \in S_i$ (strong connections).

[2] Go to next cell.

A typical example of application of this anisotropic coarsening is presented now. Starting from a C-type structured mesh around an airfoil we get the fine partition of median cells sketched in Figure 5. We observe that some vertical and horizontal lines are also stretched. Conversely, top and bottom right region are more isotropic. In the semi-coarsening partition of Figure 6, directional coarsening is evident at front/left region (two cells in one new cell); isotropic coarsening (fours cells in one new cell) can be observed in bottom of right part.

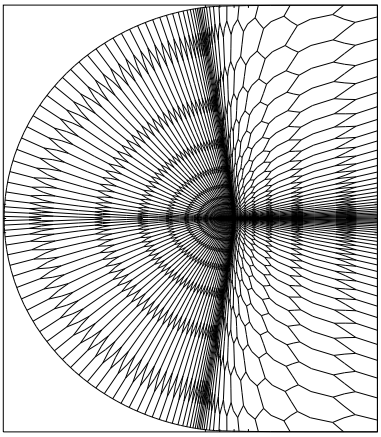

Figure 5: Fine grid : 3014 cells

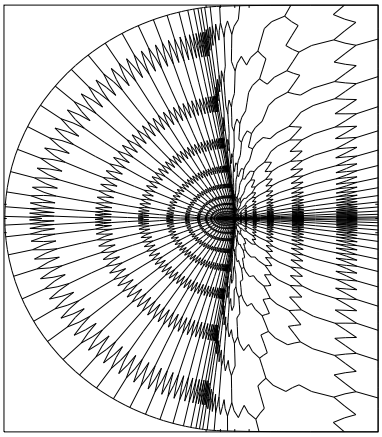

Figure 6: Coarse grid : 1380 cells

3.3 Coarse grid equation for advective term

The convectives fluxes, integrated between two control volumes of the finest grid, are computed in the same way on the coarse grid between two macro-cells. The coarse-grid matrix components are computed on the coarse grid. Both conservative variables and normal vectors are interpolated between the different grids. The normal vectors, linked at each couple of neighboring coarse macro-cells, result from the summation, once for all, of the finer grid normal vectors (for the fine cells which have a common boundary with the considered macro-cells); as a result, at most one flux is computed between two given macro-cells, i.e. the number of flux to compute per cell is comparable to the fine grid case.

3.4 Coarse grid equation for diffusive term

To evaluate diffusive terms on a coarse level, related basis functions are needed. The principle has been proposed by Koobus, Lallemand and Dervieux in [14] and consists in summing the fine grid basis functions in an algebraic equation point of view.

In the finite element formulation on the fine grid, the equations are integrated and assembled by edges and triangles. As triangles do not exist on the coarser grids, it is necessary to define a new edge-based discretization.

A variational formulation for any function f_h can be expressed by :

$$f_h(x,y) = \sum_i f_i \phi_i(x,y)$$

where f_i is the value of f_h value at a triangulation point (x_i, y_i).

The diffusive fluxes can all be written as :

$$Flux = \sum_{k=i,k\in V(i)} f_k \int_\Omega C(x,y) \frac{\partial \phi_i}{\partial x_l} \frac{\partial \phi_j}{\partial x_m} dv$$

where $C(x,y)$ contains certain characteristic values of the flow, which are averaged on edges for $i \neq j$

Let us denote:

$$L_{ij} = \int_\Omega \frac{\partial \phi_i}{\partial x_l} \frac{\partial \phi_j}{\partial x_m} dv \tag{9}$$

Integrals L_{ij} contain the basis functions gradients of the FEM formulation. They are solved on the fine grid and assembled by points and edges. A summation of these by neighboring relations on virtual coarse points, or edges connecting two macro-cells, allows one to define the coarse integrals (L_{IJ}) :

$$L_{IJ} = \int_\Omega \frac{\partial \phi_I}{\partial x_l} \frac{\partial \phi_J}{\partial x_m} dv = \sum_{i \in I, j \in J} \int_\Omega \frac{\partial \phi_i}{\partial x_l} \frac{\partial \phi_j}{\partial x_m} dv$$

This formulation is completed for isotropic and anisotropic agglomeration by a correction matrix (see [9], [14]), designed for approximatively preserving the consistency between two grids. The new feature is that directional coarsening has to be locally accounted for.
Each basis functions gradients of the $k+1$-level is multiplied by this matrix $C^I_{k,k+1}$ defined by :

$$C^I_{k,k+1} = \left(R^k_i\right)^{-1} \begin{pmatrix} C^\xi_{k,k+1} & 0 \\ 0 & C^\eta_{k,k+1} \end{pmatrix} \left(R^k_i\right) \tag{10}$$

where $k+1$ denote the indice from the coarse level (coarse grid G^{k+1} built from a fine grid G^k), R^k_i is a rotation matrix of the coordinates (x_i, y_i) and $C^\xi_{k,k+1}$, $C^\eta_{k,k+1}$ are respectively a corrective factor of the basis function gradient $\overrightarrow{\nabla} \phi_I$ following ξ and η.

For an isotropic mesh (see Figure 7 and 9, and Table 3), these correction factors are equals $C^\xi_{k,k+1} = C^\eta_{k,k+1} = C_{k,k+1}$ and defined together with the rotation matrix by :

$$C_{k,k+1} = \sqrt{2} \frac{(N_k - 1)}{(2N_k - 1)} \quad ; \quad R^k_i = I_d \tag{11}$$

where N_k is an approximation of the number of nodes in one direction (square root of the number of nodes n_k of fine grid G^k (see [14])).

For an anisotropic mesh (see Figure 11), these correction factors are put to unity in the stretching direction and $C_{k,k+1}$ in the orthogonal direction.
This device does not guarantee that the inter-grid consistency is recovered; this is in fact strictly not true; but a good part of the error is removed (see Figure 8) and the effect of this correction on convergence will be verified later (Table 5).

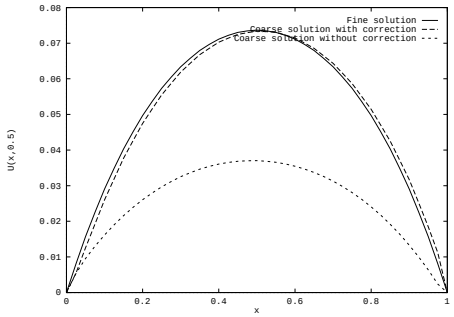

Figure 7: Anisotropic agglomeration for the Poisson problem in *a structured uniform mesh* : fine and coarse mesh solutions produced by the anisotropic coarsening with and without correction term, distribution for $y = 0.5$

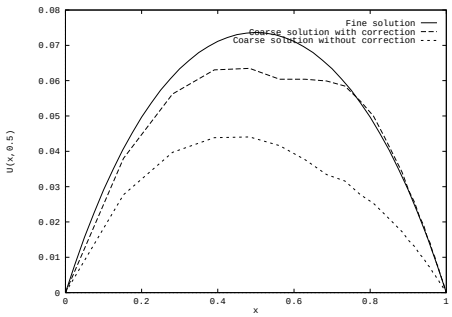

Figure 8: Anisotropic agglomeration for the Poisson problem in *a stretched structured mesh* : fine and coarse mesh solutions produced by the anisotropic coarsening with and without correction term, distribution for $y = 0.5$

3.5 Multigrid cycles

The essential features of the linear multigrid solver are standard : correction schemes formulation and V-cycle or F-cycle. We use a Gauss-Seidel iteration as a smoother. The details can be found in ([1],[9]).
The transfer operators are specific to the agglomeration techniques. We denote G^{k+1} the coarse grid built from a fine grid G^k.

- Solution restriction from fine G^k to coarse G^{k+1} is done by averaging fine solutions W^k belonging to the same coarse cell :

$$\left[R^{s}_{k,k+1}\left(W^k\right)\right]\left(C_j^{k+1}\right) = \frac{\sum_{l\in I_j^k} area\left(C_l^k\right)\ W^k\left(C_l^k\right)}{area\left(C_j^{k+1}\right)}$$

  for each coarse cell C_j^{k+1} of G^{k+1}, where I_j^k is the list of subcells, i.e. the set of level-k indices l such that $C_j^{k+1} = \bigcup_{l\in I_j^k} C_l^k$ and where $area\left(C_l^k\right)$ denotes the area of cell C_l^k of G^k.

- Residual restriction from fine G^k to coarse G^{k+1} is done by summing fine residuals res^k belonging to the same coarse cell :

$$\left[R^{rhs}_{k,k+1}\left(res^k\right)\right]\left(C_j^{k+1}\right) = \sum_{l\in I_j^k} res^k\left(C_l^k\right)$$

  for each coarse cell C_j^{k+1} of G^{k+1}.

- Correction prolongation from coarse (G^{k+1}) to fine (G^k) is composed of a trivial injection :

$$\left[P_{k+1,k}\left(e^{k+1}\right)\right]\left(C_j^k\right) = e^{k+1}\left(C_l^{k+1}\right)$$

for each fine cell C_j^k of G^k, where l is the index of the coarse cell C_l^{k+1} which contains fine cell C_j^k.

4 Application to linear scalar problem

We first consider the application of a linear MG cycle built from the above options for solving the Poisson problem on several types of mesh with Dirichlet condition on the boundary. The approximation is purely a P1-Galerkin one.

4.1 Structured uniform mesh

A first mesh is a square one by one and not stretched. The mesh contains 41×41 points (see Figure 9). A rotation was applied to avoid alignment of mesh with cartesien coordinates.
We observe in Table 2 that convergence is as good with both algorithms which build meshes not much different, since the problem is isotropic.

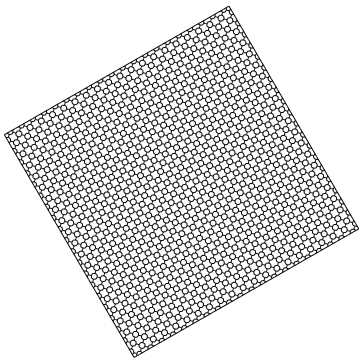

Figure 9: Fine mesh : 1681 cells

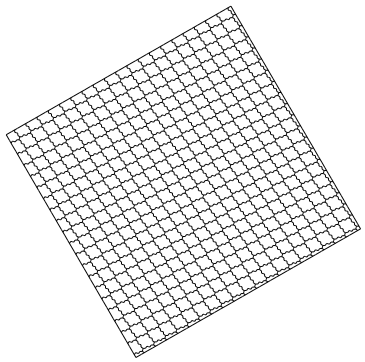

Figure 10: Coarse grid : 441 cells

Cycles	*Isotropic MG*				*Anisotropic MG*			
	nivx	μ_{moy}	α_6	*cplx*	*nivx*	μ_{moy}	α_6	*cplx*
BGI(2,2)	2	0.202	10	∞	2	0.166	9	∞
V-cycle(2,2)	5	0.227	11	67	5	0.177	9	56
F-cycle(2,2)	5	0.175	9	68	5	0.159	9	70

Table 2: Poisson equation solution on a uniform mesh: comparison between isotropic and anisotropic coarsening; α_6 denotes the number of cycles needed for a 6-decade residual reduction

Cycles	*Anisotropic MG (without)*				*Anisotropic MG (with)*			
	nivx	μ_{moy}	α_6	*cplx*	*nivx*	μ_{moy}	α_6	*cplx*
BGI(2,2)	2	0.510	22	∞	2	0.166	9	∞
V-cycle(2,2)	5	0.800	64	392	5	0.177	9	56
F-cycle(2,2)	5	0.669	36	273	5	0.159	9	70

Table 3: Poisson equation solution on a uniform mesh: influence of the correction factor (with or without)

4.2 Stretched structured mesh

The mesh is identical of first mesh but with a geometrical progression according to (Ox) (see Figure 11). The maximal aspect ratio is 100.
From Table 4, we observe that the isotropic algorithm has real troubles. The anisotropic one has performance not much worse than in the previous isotropic test case.

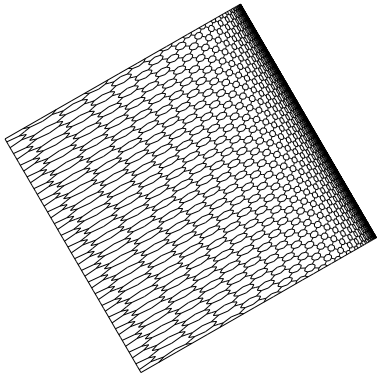

Figure 11: Stretched fine mesh : 1681 cells

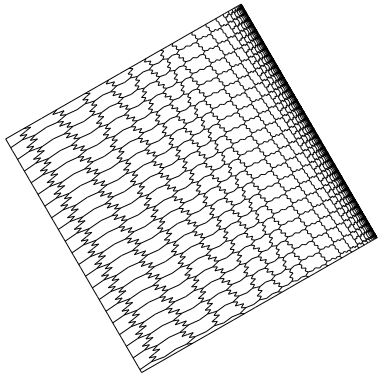

Figure 12: Coarse grid : 761 cells

Cycles	*Isotropic MG*				*Anisotropic MG*			
	nivx	μ_{moy}	α_6	*cplx*	*nivx*	μ_{moy}	α_6	*cplx*
BGI(2,2)	2	0.796	62	∞	2	0.201	10	∞
V-cycle(2,2)	5	0.801	64	391	6	0.224	11	87
F-cycle(2,2)	5	0.798	63	476	6	0.227	11	130

Table 4: Poisson equation solution on a stretched mesh: comparison between isotropic and anisotropic coarsening; α_6 denotes the number of cycles needed for a 6-decade residual reduction

Cycles	*Anisotropic MG (without)*				*Anisotropic MG (with)*			
	nivx	μ_{moy}	α_6	*cplx*	*nivx*	μ_{moy}	α_6	*cplx*
BGI(2,2)	2	0.454	19	∞	2	0.201	10	∞
V-cycle(2,2)	6	0.752	50	401	6	0.224	11	87
F-cycle(2,2)	6	0.615	30	367	6	0.227	11	130

Table 5: Poisson equation solution on a stretched mesh: influence of the correction factor (with or without)

4.3 "Flat plate" mesh

We consider a mesh usually used for calculations of a turbulent flow on a flat plate (see Figure 18; the maximum aspect ratio is 5000. We examine the iterative solution of the Poisson equation. The effect of such a stretching on the standard MG approach with the regular coarsening is

obvious : we observe a loss in convergence speed and the reduction factor is .90 (see Table 6). Contrarily, the anisotropic semi-coarsening strategy proves again to be much less sensitive to stretching (reduction factor is .24; Table 6). A gain the correction factor shows its influence (Table 7).

Cycles	*Isotropic MG*				*Anisotropic MG*			
	nivx	μ_{moy}	α_6	*cplx*	*nivx*	μ_{moy}	α_6	*cplx*
BGI(2,2)	2	0.902	135	∞	2	0.191	10	∞
V-cycle(2,2)	5	0.903	137	827	6	0.384	16	132
F-cycle(2,2)	5	0.906	141	1041	6	0.239	11	141

Table 6: Poisson equation solution on a flat plate mesh: comparison between isotropic and anisotropic coarsening; α_6 denotes the number of cycles needed for a 6 decade residual reduction

Cycles	*Anisotropic MG (without)*				*Anisotropic MG (with)*			
	nivx	μ_{moy}	α_6	*cplx*	*nivx*	μ_{moy}	α_6	*cplx*
BGI(2,2)	2	0.472	20	∞	2	0.191	10	∞
V-cycle(2,2)	8	0.817	70	588	6	0.384	16	132
F-cycle(2,2)	8	0.683	38	522	6	0.239	11	141

Table 7: Poisson equation solution on a flat plate mesh: influence of the correction factor (with or without)

4.4 Some comments

Performances for the diffusion-advection model are generally better than for the Poisson case (see [14],[9] for details). Although more costly in the above experiments, the F-cycle appears to have a convergence less sensitive to test cases. However the V-cycle will be used in the rest of the paper.

5 Extension to Navier-Stokes

5.1 Physical model

The governing equations are obtained by Reynolds averaging the compressible Navier-Stokes equations, and modeling the Reynolds stress by the Boussinesq assumption.
These equations can be written in a conservative form as:

$$\frac{\partial W}{\partial t} + \frac{\partial F(W)}{\partial x} + \frac{\partial G(W)}{\partial y} = \frac{1}{Re}\left(\frac{\partial R(W)}{\partial x} + \frac{\partial S(W)}{\partial y}\right) + \frac{\partial \tilde{R}(W)}{\partial x} + \frac{\partial \tilde{S}(W)}{\partial y} + \Omega(W)$$

where

- $W(x, y, t)$ is vector function of $I\!R^6$, the components of which are the non-dimensional conservative variables
- $F(W)$ et $G(W)$ are the functions of the convective fluxes
- $R(W)$, $S(W)$ are the functions of laminar viscous fluxes and Re is the laminar Reynolds numbers from nquation non-dimension
- $\tilde{R}(W)$, $\tilde{S}(W)$ are the functions of turbulent viscous fluxes
- $\Omega(W)$ is the source term of the turbulence

The closure of the system is realized by the high-Reynolds number $k - \varepsilon$ turbulent model of Launder-Spalding [17]. In order to treat the hyperbolic terms in the same way as in the laminar case, an appropriate change of variables is used. One lets:

$$\begin{cases} p^{'} = p + \dfrac{2}{3}\rho k \\ E^{'} = E + \beta\rho k \quad \text{and} \quad \beta = -1 + \dfrac{2}{3(\gamma - 1)} \end{cases}$$

where

$$\begin{cases} E = \rho C_v T + \dfrac{1}{2}\rho(u^2 + v^2) + \rho k \\ p = (\gamma - 1)\rho C_v T, \qquad \gamma = 1.4 \end{cases}$$

where p is the pressure, E the total energy per unit volume, ρ the fluid density, k the turbulent kinetic energy, C_v denotes the specific heat at constant volume, T the temperature and u, v are the component of fluid velocity.
The relation between $E^{'}$ and $p^{'}$ written the classical way as:

$$p^{'} = (\gamma - 1)\left(E^{'} - \frac{1}{2}\rho\left(u^2 + v^2\right)\right).$$

Thus, the change of variables intruduced in the initial system allows to keep the classical conservative form of the physical variables.

We denote respectively by μ_t and ω_k, ω_ε, the turbulent viscosity and the components source term for k, ε :

$$\begin{cases} \mu_t = c_\mu \dfrac{\rho k^2}{\varepsilon} \\ \\ \omega_k = -\rho\,\varepsilon + \mathcal{P} \\ \\ \omega_\varepsilon = c_{\varepsilon_1} \dfrac{\varepsilon}{k}\,\mathcal{P} - c_{\varepsilon_2} \dfrac{\rho\,\varepsilon^2}{k} \end{cases}$$

where

$$\mathcal{P} = -\left(\frac{2}{3}\rho k \delta_{ij} - \mu_t \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right)\right)\frac{\partial u_i}{\partial x_j}$$

with c_μ, c_{ε_1} and c_{ε_2} are the empirical constants of the modeling, ε denote the turbulent dissipation rate and $\mathcal{P}$ the production term of the turbulent. The laminar Prandtl number is $Pr = 0.725$ and the turbulent Prandtl number is $Pt = 0.86$.

The above transport equations on k and ε are obtained by assuming that the turbulent effects are dominant in the flow domain [12]. Thus the standard $k-\varepsilon$ model is not valid in regions where the viscous effects are large compared with the turbulent effects (near-wall zones). In order to account for low-Reynolds number effects (near-wall effects), Chen-Patel [3] (1988) proposed to solve the $k-\varepsilon$ equations only in the high-Reynolds number regions and to use a low-Reynolds number one-equation model in the near-wall region. In many cases one-equation models show near the wall a better agreement with experiments than standard two-equation models [10]. A further advantage is that one-equation models require less mesh nodes in the viscous sublayer than low-Reynolds two equations models and this increases the computational efficiency and convergence properties of the numerical method.
Following the approach of Chen and Patel [3], the one-equation low-Reynolds number model of Wolfshstein [23] has been included in the present method. In regions adjacent to the surface, where $R_y < 200$, the mean-flow equations and the equation for the turbulent kinetic energy are solved, whilst the characteristic length scales are determined via algebraic relations. In these regions, the eddy viscosity employed is defined by

$$\mu_t = c_\mu \;\; \rho\,\sqrt{k}\; f_\mu, \quad f_\mu = \sqrt{\frac{\rho_w}{\rho}}\; l_\mu$$

where ρ_w is the fluid density of the wall and the dissipation rate of turbulent kinetic energy employed for the modelled source terms in the equation for k is given by

$$\varepsilon = \frac{k^{\frac{3}{2}}}{l_\varepsilon}$$

with l_μ and l_ε are two algebraic length scales, defined in the following way

$$l_\mu = C_1\; y\;\left(\;1 - \exp\left(\frac{-R_y}{A_\mu}\right)\right) \qquad l_\varepsilon = C_1\; y\;\left(\;1 - \exp\left(\frac{-R_y}{A_\varepsilon}\right)\right).$$

The damping functions, which contain the turbulent Reynolds number R_y, are introduced to mimic the correct behaviour when the wall is approached and the flow is totally dominated by effects due to the molecular viscosity. The model requires two constants which are

$$A_\mu = 70, \quad C_1 = \kappa \; {c_\mu}^{-3/4}, \quad R_y = \sqrt{\frac{\rho}{\rho_w}} \frac{\sqrt{k}\; y}{\nu_w}$$

where κ and c_μ take the usual values of 0.45 and 0.09.

A detailed discussion of the merit of this model compared with genuine low-Reynolds two-equations models is presented in [10].

5.2 Implicit time advancing

It is well known that a global Newton iteration starting from an arbitrary initialization cannot be applied to a compressible flow. In order to approach the convergence domain of a (modified) Newton iteration, an Euler backward-time linearised implicit advancing is constructed, with the following features:

- Linearization : except for the production part in the source term, the linearization is obtained by freezing the Jacobian in Roe's flux difference splitting for convective terms, and by exact differentiation of diffusion and source terms. Also, the turbulent viscosity μ_t is frozen so that the k and ε variables are coupled to each other but uncoupled from the other four flow variables.

- A Roe approximate Riemann solver is used for the approximation of the fluid convective terms; the positivity- preserving multi-component Riemann flux proposed in [16] is considered for the turbulence convective terms.

- Preconditioning : it is performed using a first-order Godunov scheme because it is tridiagonal in 1D and better conditioned than the second-order one, which furthermore produces larger matrices (pentadiagonal in 1D).

- Local time stepping and time-step incrementation : a local time-step is computed on each cell so that the Courant number (denoted by "CFL" in the sequel) is somewhat uniform on the mesh. The CFL number is an increasing function of time and of the nonlinear residual L_2 norm, in order to ensure the progressive switch from the unsteady phase to the asymptotic convergence.

- Each time step involves the solution of the two linear systems (mean flow and closure variables) by an agglomeration multi-grid method described in next subsection.

- Each computation is started from a uniform flow.

5.3 Multigrid solution of the linear system

The linearised system to be solved is essentially of the same type as advection-diffusion. Its MG treatment is proposed in [1] and we recall now some main features of that approach and some differences with [1].

Main features : coarse grid systems rely on Finite-Volume for the (first-order acurate) advective terms and on integrals L_{ij} (see (9)) for other terms. Variable coefficients arising from the

freezing of flow field are derived from restriction of the flow field variables, and edgewise constant values are taken for simplifying the building of edge based diffusion terms.
Novelties with respect to [1] : several important quantities are transfered to coarse grids : friction velocity, normal distance to wall, turbulent local Reynolds number R_y, wall laminar viscosity ν_t. The reduction of unknows in the layer obtained by putting to zero non-diagonal terms for ε, and fixing the diffusion of ε in (block) diagonal terms.
The V-cycle with 2 sweeps per level is used. Only one cycle per time step is performed.

6 Application to the flows problems

6.1 Laminar flow around an airfoil

We now consider the introduction of the semi-coarsening method in the laminar Navier-Stokes solver presented in Section 3.2.2. The farfield Mach number is 0.8, the angle of attack is 10 deg., the Reynolds number is 73. The mesh used (involving again about 12000 nodes) is stretched near the airfoil and in other places according to a C-type structured topology (see Figure 14). Figure 15 depicts the pseudo-time nonlinear convergence of the implicit scheme when it is equipped by the present MG scheme. When the isotropic version is used, the convergence and the efficiency is better than with a single-grid equivalent approach, but its convergence in 800 iterations (for ten decades) makes it rather disappointing as compared to the computation done with a non-stretched mesh. The anisotropic semi-coarsened version does much better with a convergence in 140 iterations and is four time more efficient in terms of computer time (see Table 9).

DEC Alpha 266	Iso. MG	Aniso. MG
CPU time	63 min.	12 min.

Table 8: CPU time following using the MG methods

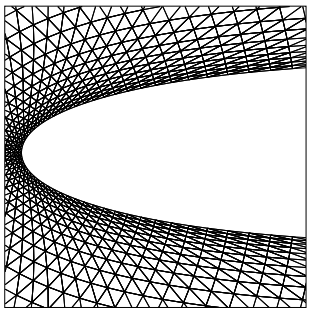

Figure 13: Zoom of the airfoil mesh near leading edge

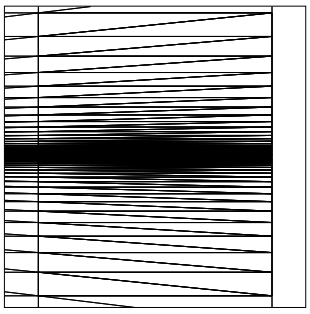

Figure 14: Zoom of the mesh on a region of the wake

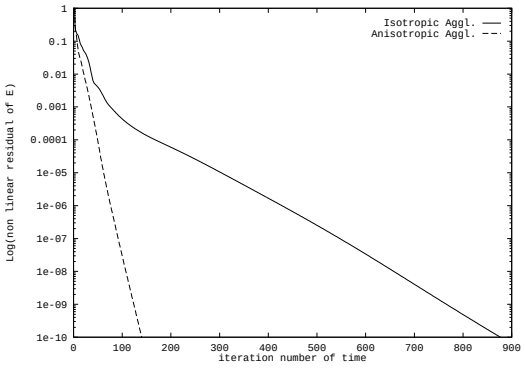

Figure 15: Laminar flow past a NACA airfoil: convergence history for the isotropic MG (800 iterations) and the anisotropic one (140 iterations)

6.2 Turbulent supersonic flow over a flat plate

A second test case is the computation of a turbulent flat plate flow. The case was introduced and experimented by Mabey (see [8]; farfield Mach number is 4.52 and Reynolds number is 28.2 10^6 m^{-1}); this flow was chosen recently as a test case for the ETMA Workshop [7] for which a common mesh involving 113×81 nodes was provided by Vrije University of Brussels (thanks to Ch. Hirsch and E. Shang); we use this mesh for which the first node row corresponds to a Y^+ less than 1; the mesh is highly stretched with an aspect ratio near wall of 5000. The turbulence model chosen is a $k-\epsilon$ one with a two-layer Chen-Patel [3] treatment. In Figures 16 and 17 we give an idea of the weak level of model and numerical errors (mesh convergence has been verified as good, except maybe at proximity of the uniform flow, see [8]). In Figure 21 are presented three convergence histories, obtained with an implicit time advancing in which the linearized system is solved by three different algorithms, but with about the same CPU time per time iteration. The slower convergence is obtained with a few linear Jacobi (Single-Grid) sweeps for each time step; after a rather good phase, convergence slowens to a fair rate. The second convergence results from the application of a MG cycling relying on isotropic coarsening; according to theory, MG should not be much better than Single-Grid on this very stretched; a kind of confirmation is given by convergence: after a phase converging better than Single-Grid, convergence also degrades to an analogous asymptotic rate, proving that, for example, in a 4-decade converged solution, a component of the error where not damped by a factor 10. Conversely, a quasi-constant convergence rate is obtained with the new anisotropic algorithm. Futher, the convergence is essentially not sensitive to mesh size (Figure 22). Note that mesh convergence is observed toward a solution that has a pretty good agreement with the measurements of Mabey, for the two typical quantities that are streamwise friction at wall and distribution of x-moment normal to the wall (see Figure 16 and 17).

DEC Alpha 266	Single-Grid	Iso. MG	Aniso. MG
CPU time	65min.	42 min.	13 min.

Table 9: CPU time following using the MG methods

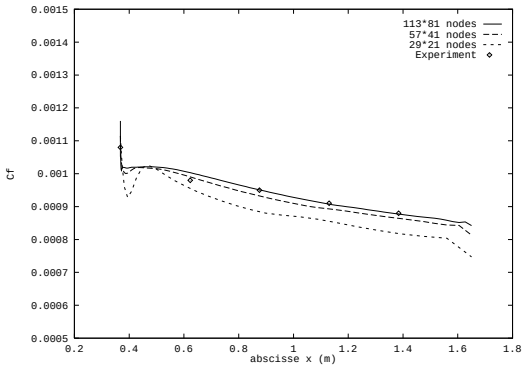

Figure 16: Mabey flat plate test case for 29×21, 57×41, 113×81 meshes : frictions

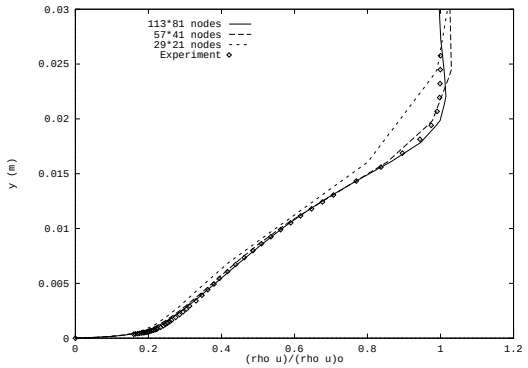

Figure 17: Mabey flat plate test case for 29×21, 57×41, 113×81 meshes : x-moment on $x = 1.384$

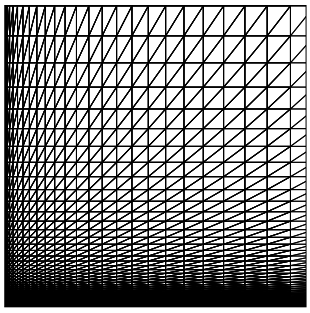

Figure 18: Zoom (bottom left) of the flat plate mesh : 113 × 81 points

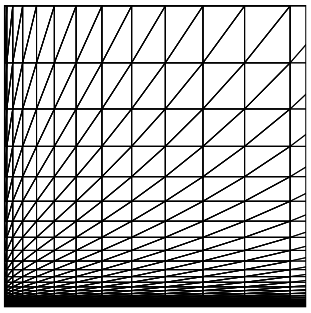

Figure 19: Zoom of the flat plate mesh : 57 × 41 points

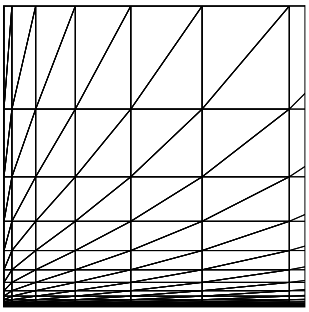

Figure 20: Zoom of the flat plate mesh : 29 × 21 points

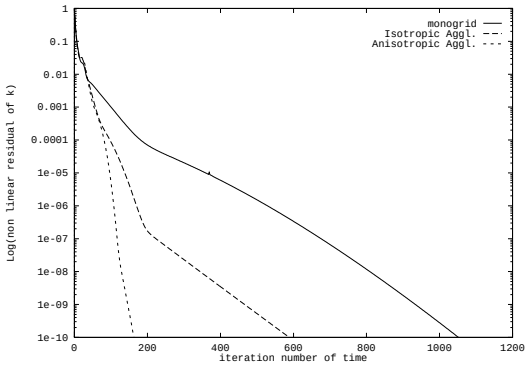

Figure 21: Turbulent flow past a plate flat: convergence history for the single grid implicit algorithms (1050), isotropic MG (580 iterations) and the anisotropic MG (180 iterations)

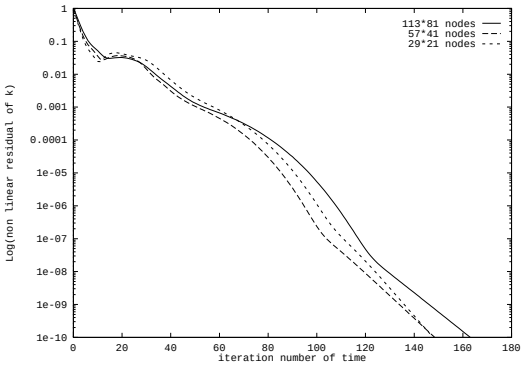

Figure 22: Turbulent flow past a plate flat for 29×21, 57×41, 113×81 meshes : convergence histories

6.3 Turbulent subsonique flow around an airfoil

A much more difficult case is now considered: it consist of the flow around an OALT25 airfoil. The geometry has the particularity of being finished as a truncated forebody (Figure 24). The mesh (courtesy of Dassault-Aviation) is of hybrid type with an unstructured part far from wall and a structured layer near wall. The total number of vertices is 19 756.

In order to obtain a realistic shock location, the above model has ben modified according to the Menter method [19]. This method is based on the one equation model of Johnson-King [13] :

$$\mu_t = \frac{\rho\ a\ k}{\max\left(\frac{a\ \varepsilon}{c_\mu\ k}, \left|\frac{\partial u}{\partial y}\ F\right|\right)} \tag{12}$$

where

$$F = \begin{cases} 1 & \text{si } y^+ < 1000 \\ & \\ 0 & \text{sinon} \end{cases} \qquad \text{and} \quad a = \sqrt{c_\mu} \tag{13}$$

Smaller time steps are used in this case and the improvement of convergence and efficiency is less impressive (Table 10, Figure 27 and 28). Pressure distribution has some agreement with experiment (see Figure 25).

DEC Alpha 266	Single-Grid	Aniso. MG
CPU time	68 min.	49 min.

Table 10: CPU time following using the methods

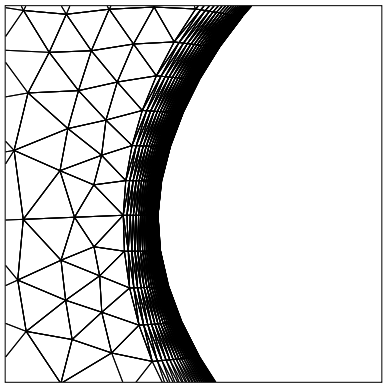

Figure 23: Zoom of the airfoil mesh

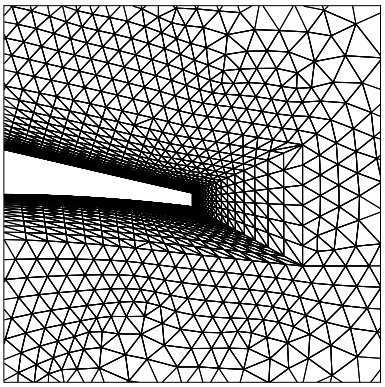

Figure 24: Zoom trailing edge

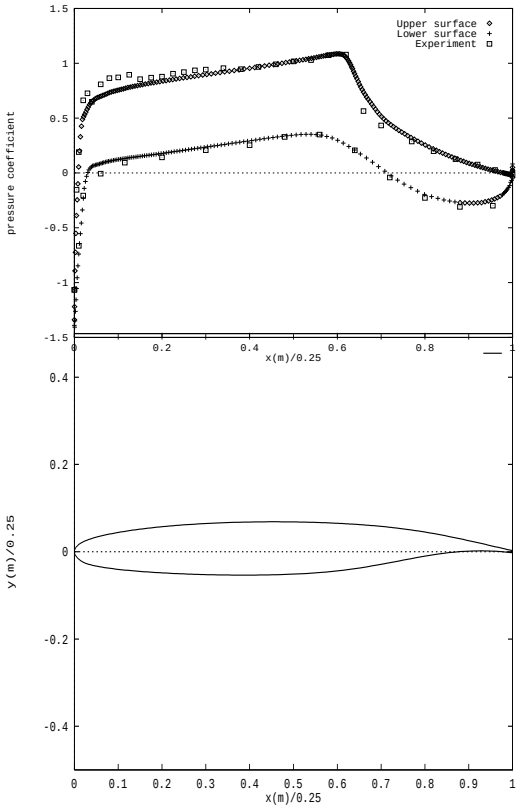

Figure 25: Turbulent flow past a OALT25 airfoil (Mach 0.7817) : comparaison of computed pressure coefficient with experimental measurements

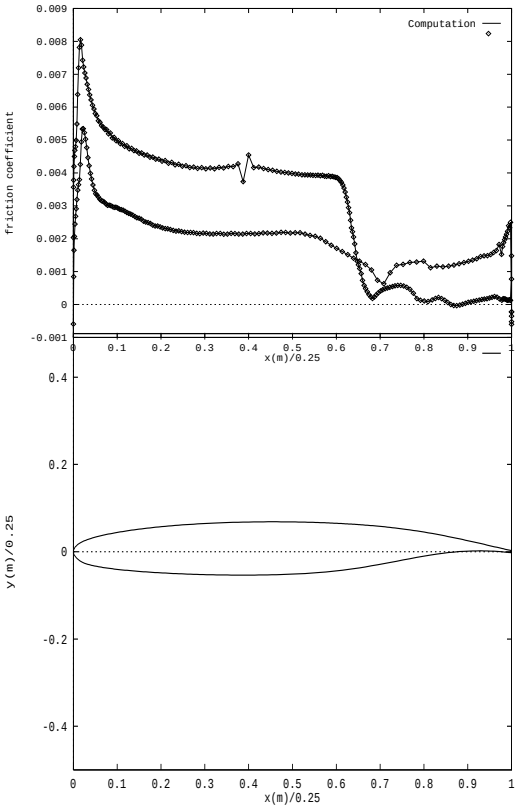

Figure 26: Turbulent flow past a OALT25 airfoil (Mach 0.7817) : friction coefficient

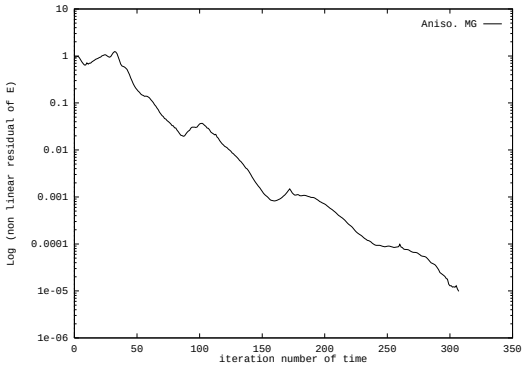

Figure 27: Turbulent flow past a OALT25 airfoil (Mach 0.7817) : convergence history of anisotropic MG (300 iterations)

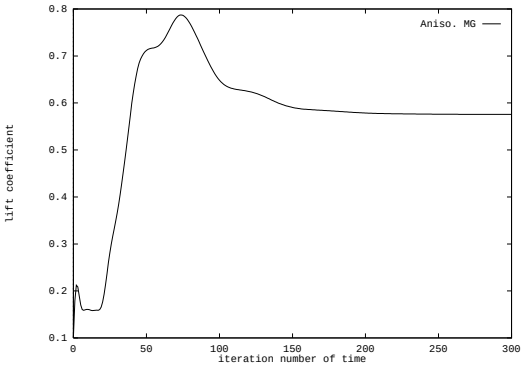

Figure 28: Turbulent flow past a OALT25 airfoil (Mach 0.7817) : lift coefficient as a function of time-iteration

7 Conclusion

The main question addressed in this paper is whether MG methods can perform on stretched unstructured meshes as well as they do on isotropic meshes. Indeed, the theory seems to indicate that convergence should be at least as good for an anisotropic algorithm applied to a stretched mesh as for an isotropic algorithm applied to an isotropic (non-stretched) mesh. In other words, the only source of additional cost should be the fact that semi-coarsened meshes contain more nodes than fully coarsened ones.

In the above study, we got experimental confirmation that the isotropic algorithm, that performs well for isotropic meshes, has a much slower convergence when applied to stretched meshes, for models problems as Poisson, as well as complex compressible flows. In some case, asymptotic convergence is not significantly better than for the analogous single grid scheme.
We have proposed an isotropic algorithm for which experiments show the expected improvement:
For the elliptic model problem, convergence becomes insensible to mesh stretching: while about 9-10 cycles are used for the isotropic MG on non-stretched meshes, the anisotropic MG needs also 10-12 cycles for the same computation, but also about 10-12 cycles for a stretched-mesh computation.
For the compressible flows presented, the poorer asymptotic convergence rate of the isotropic version is not observed for the anisotropic one; at the contrary, convergence rate is rather constant from beginning to end of convergence. This last point is essential, the slower mode is generally not solved in a strategy in which convergence is stopped after some cycles of quasi-stagnation. The fact that this happens for an already small residual level can be deceitful, especially when shock and boundary layers can interact, and may yield bad predictions.
In other cases, it may even happen than the slower error mode be large at initial conditions and the bad asymptotic rate may appear early.

However the presented algorithm may not enjoy this ideal property for all cases, and further experiments and improvement are necessary to qualify it as a kind of ultimate anisotropic MG method in 2D.

Also most experiment were done with locally structured stretched meshes because good unstructured highly streched meshs are still difficult to build.

For 3D extension, although no explicit use of the two-dimensionality is done in building the method, we anticipate some -hopefully resolvable- extra difficulties in the derivation of a semi-coarsening algorithm ensuring efficiency for most types of meshes.

8 Acknowledgements

The authors would like to thank B. Koobus, G. Carré and H. Guillard from INRIA sophia-Antipolis for their valuable advices and suggestions.

The authors would also like to acknowledge M. Ravachol from Dassault-Aviation, F. Cantariti from University of Glasgow and Erbing Shang from University of Brussel for the fruitful discussions they had with them.

References

[1] G. Carré, " An Implicit Multigrid Method by Agglomeration Applied to Turbulent Flows "- *International Journal of Fluids Engineering*, (1996)

[2] G. Carré, A. Dervieux, and J. Francescatto, "On anisotropic MG and on FMG algorithms for Navier-Stokes analysis on unstructured meshes", *In J.A. Desideri & P. Le Talleç editor*, Proc. of ECCOMAS 1996. John Wiley & Sons

[3] H.C. Chen and V.C. Patel, "Near-Wall Turbulence Models for Complex Flows Including Separation", *AIAA J.*, Vol. 26, pp. 641-648 (1988)

[4] H. Deconinck, R. Strujis, K.G. Powell and P. Roe, *AIAA Paper 91-1532*, AIAA 10th Computational Fluid Dynamics Conference, Honolulu, Hawaii, (1991)

[5] A. Dervieux, "Steady Euler Simulations Using Unstructured Meshes" - *Von Karman Institute Lecture Series* 85-04 (1985). *Published in Partial Differential Equations of Hyperbolic Type and Applications, G. Geymonat, Ed., World Scientist* pp. 34-105 Singapore (1985)

[6] A. Dervieux, J. Francescatto and G. Carré, "Fast Solver for Unstructured Finite Volume Methods", *In First International Symposium on Finite Volumes for Complexes Applications*, Rouen (July 1996)

[7] A. Dervieux, J.-P. Dussauge (eds.) "Computation and Comparison of Efficient Turbulence Models for Aeronautics", *Proceeding of the ETMA Workshop*, to be published by Vieweg

[8] P.J. Finley, H.H. Fernholz "A Critical Compilation of Compressible Turbulent Boundary Data", *AGARDOgraph*, 223 (1977)

[9] J. Francescatto " Résolution de l'équation de Poisson par une méthode multigrille agglomeré sur maillages étirés", *INRIA Research Report*, N^o 2712 (1995)

[10] J. Francescatto "Modèles bas-Reynolds appliqués à une couche limite compressible", *INRIA Research Report*, No 2837 (1996)

[11] P.L. George, F. Hecht and E. Saltel, *Comput. Methods. Appl. Mech. Engrg. 33*, 975-995, (1991)

[12] W.P. Jones, B.E. Launder, "The Prediction of Laminarization with a Two-Equation Model of Turbulence", *Int. J. Heat Mass Transfer*, 15, 301-314 (1972)

[13] D.A. Johnson, L.S. King, "A Mathematically Simple Turbulence Closure Model for Attached and Separated Turbulent Boundary Layers", *AIAA Journal*, Vol 23, p. 1684-1692, (1985)

[14] B. Koobus, M-H Lallemand and A. Dervieux "Unstructured Volume-Agglomeration MG : Solution of the Poisson Equation",*INRIA Research Report*, N^o 1946 (1993)

[15] M-H. Lallemand, H. Steve and A. Dervieux ,"Unstructured Multigridding by Volume Agglomeration : Current Status", *Computers and Fluids*, Vol.21, pp. 397-433 (1992)

[16] B. Larrouturou, "How to Preserve the Mass Fraction Positivity when Computing Compressible Multi-Component Flows"- *J. Comput. Phys.*, Vol 1., pp. 59-84 (1991)

[17] B.E. Launder and D.B. Spalding, "The Numerical Computation of Turbulent Flows", *Computer Methods in Applied Mechanics and Engineering*, Vol.3, pp. 269-289 (1974)

[18] D.J. Mavriplis, V. Venkatakrishnan, "A Unified Multigrid Solver for the Navier-Stokes Equations on Mixed Element Meshes", *ICASE Report* 95-53 (1995)

[19] F.R. Menter, "Zonal Two-Equation $k - w$ Turbulence Models for Aerodynamic Flows", *AIAA 93-2906*, (1993)

[20] K. Mer, "Variational Analysis of a Mixed Finite Element Finite Volume Scheme on General Triangulations", *INRIA Research Report*, N^o 2213 (1994)

[21] E. Morano, D.J. Mavriplis and V. Venkakrishnan "Coarsening Strategies for Unstructured Multigrid Techniques with Application to Anisotropic Problems", *ICASE Report* No. 95-34 (1995)

[22] A.W. Mulder, "A New Multigrid Approach to Convection Problems", *J. Comput. Phys.*, Vol. 83, pp. 303-323 (1989)

[23] M. Wolfshtein "The Velocity and Temperature Distribution in One-Dimensional Flow whith Turbulence Augmentation and Pressure Gradient", *International Journal of Heat and Mass Transfer*, Vol. 12, pp. 301-318 (1969)

Unité de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unité de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unité de recherche INRIA Rhône-Alpes, 655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN
Unité de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unité de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
ISSN 0249-6399